Intro to Big Data and Hadoop

What Is Big Data?

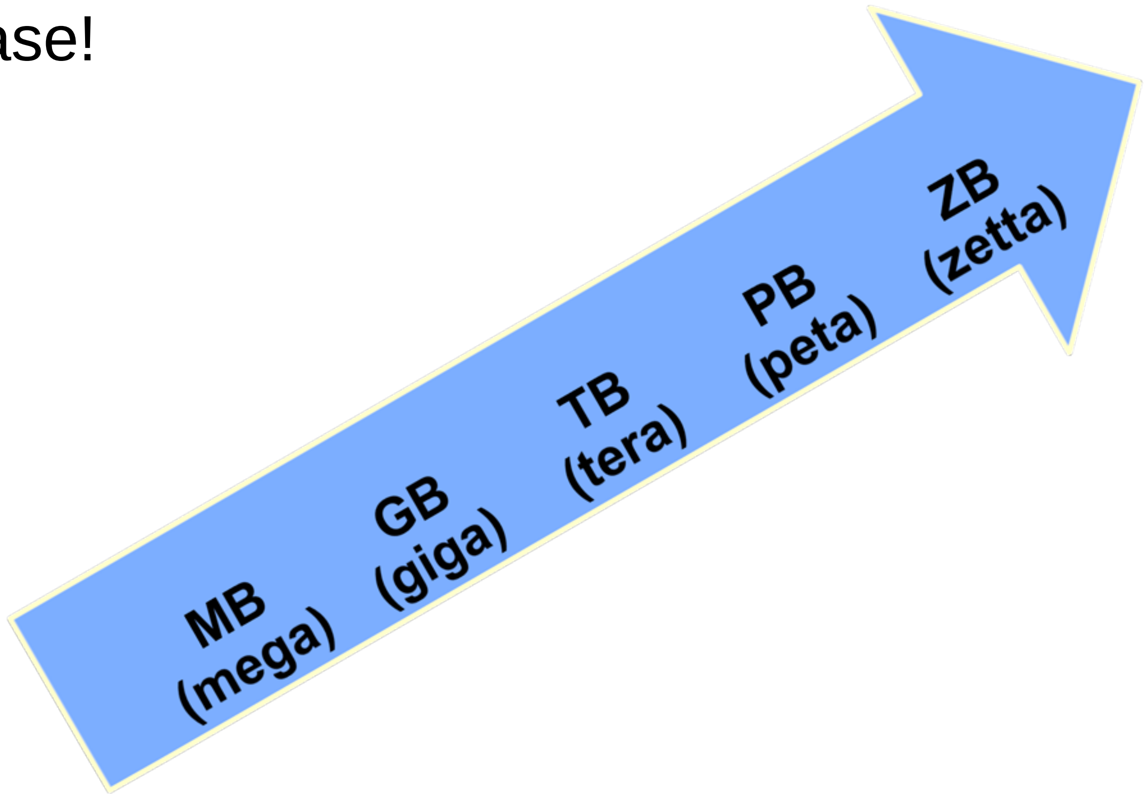
Definitions for *big data* found on Wikipedia:

"Big data is a collection of data sets so large and complex that it becomes difficult to process using on-hand database management tools."

"Big data is a term used to describe data that cannot be managed and analyzed using traditional infrastructure, architecture, and technologies."

Disk Size Prefixes

The amount of data being stored and analyzed continues to increase!

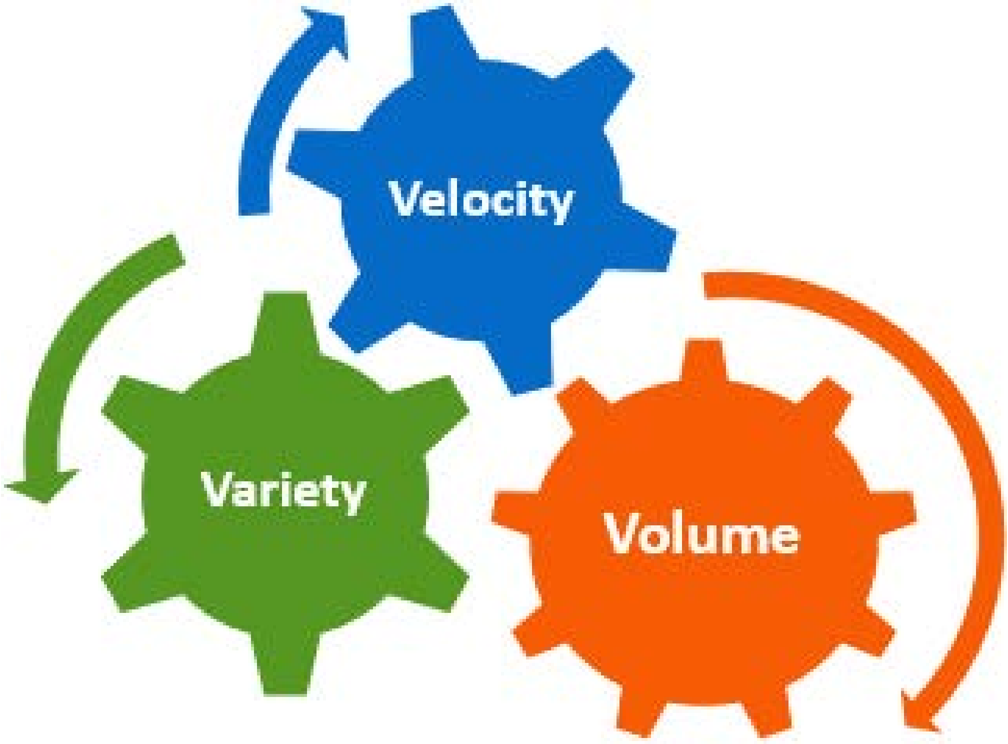


A good type rate is 10 KB/hour (0.01 MB/hr). An E-book is ~ 1 MB. A typical disk drive write rate is ~100-250 MB/sec (100 GB in 20 minutes; 1 TB in 3 hours, 1 PB in 4 months). Good internet is similar.

Attributes of Big Data

The following three characteristics make data “big data”:

- ✓ **Volume** (Terabytes -> Petabytes)
- ✓ **Velocity** (Batch -> Streaming Data)
- ✓ **Variety** (Structured -> Unstructured)



Big Data Estimates for Three Vs

Volume

- >3500 petabytes in North America
- >3000 petabytes for rest of the world

Variety

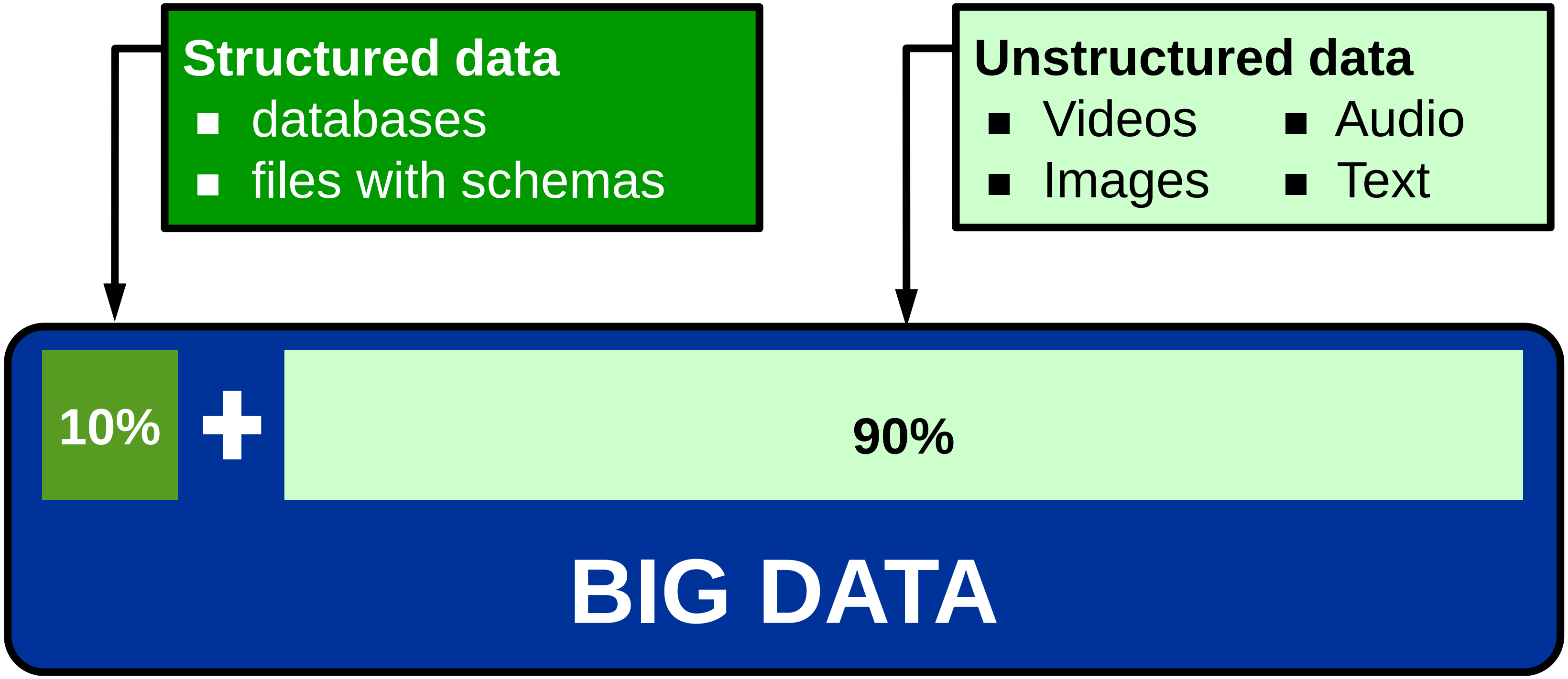
- People (*web*, *social media*, *e-commerce*, *music*, *videos*, *messaging*)
- Machines (*sensors*, *medical devices*, *GPS*)

Velocity

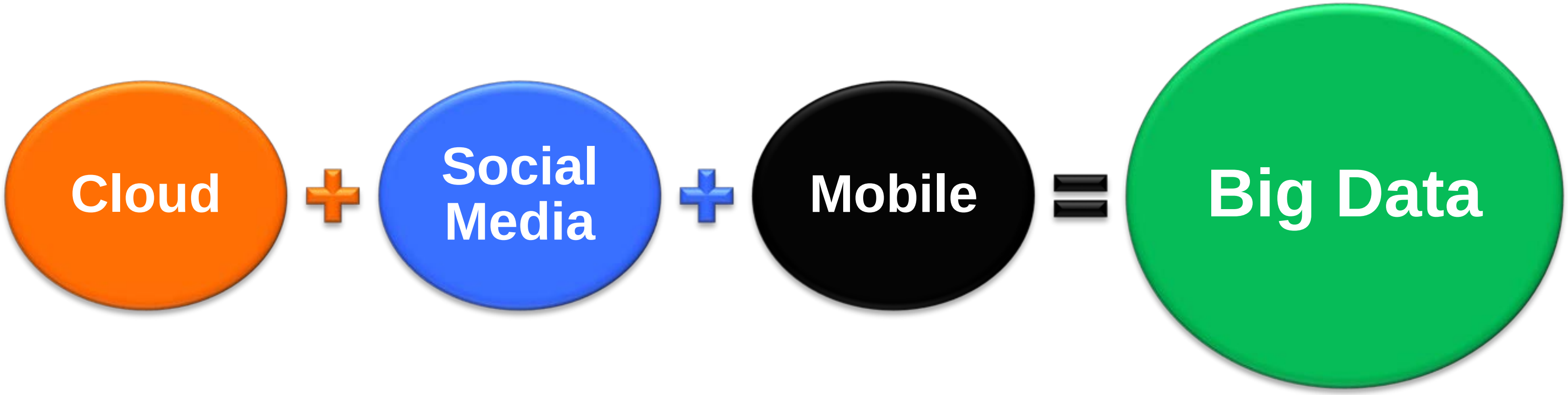
- About 3 million emails per second
- 200,000 logins on Facebook every minute
- Millions of stock trades in seconds

Big Data Variety

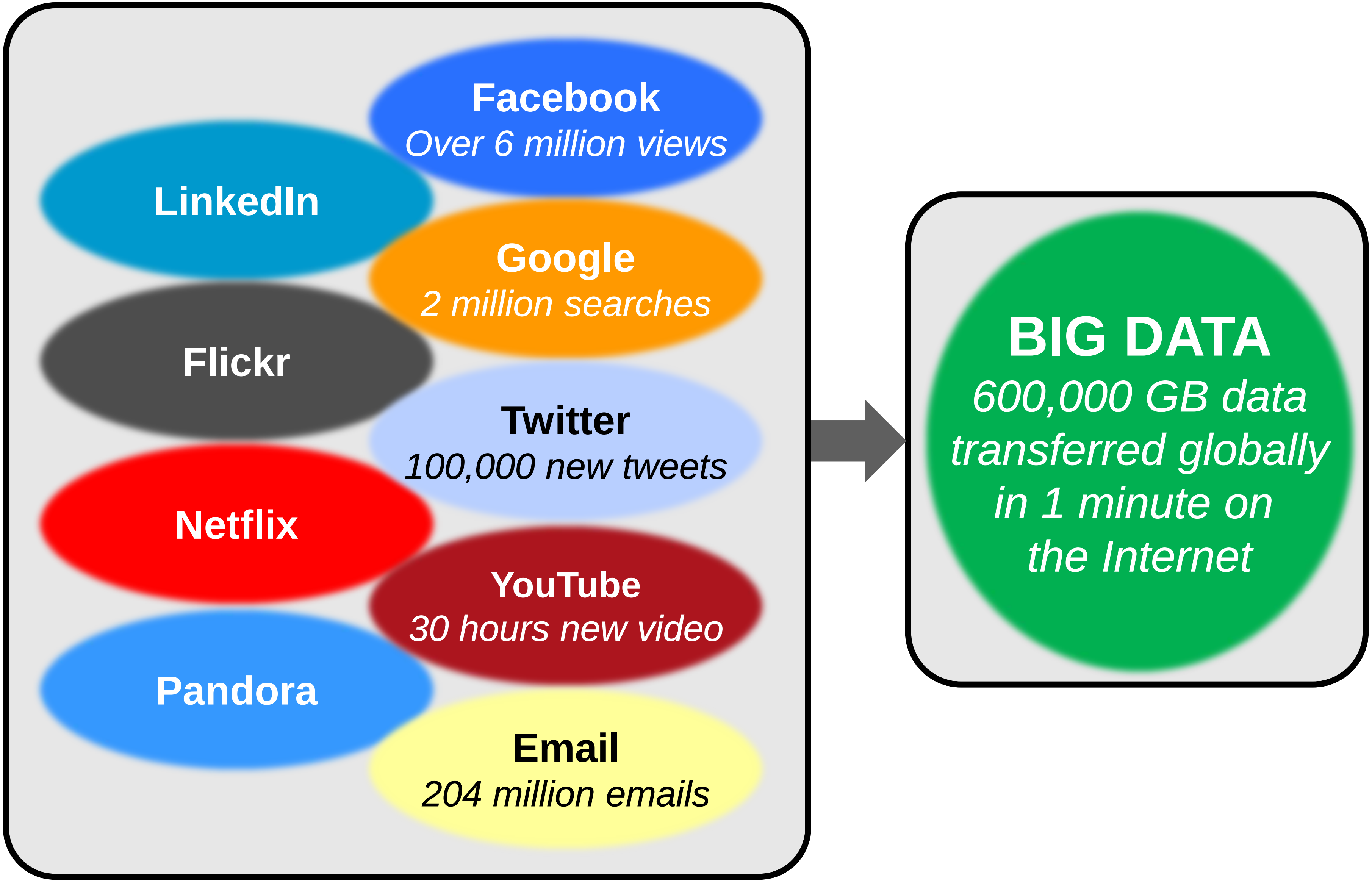
Big data consists of structured and unstructured data. It is estimated that almost 90% of “big data” comes from unstructured data.



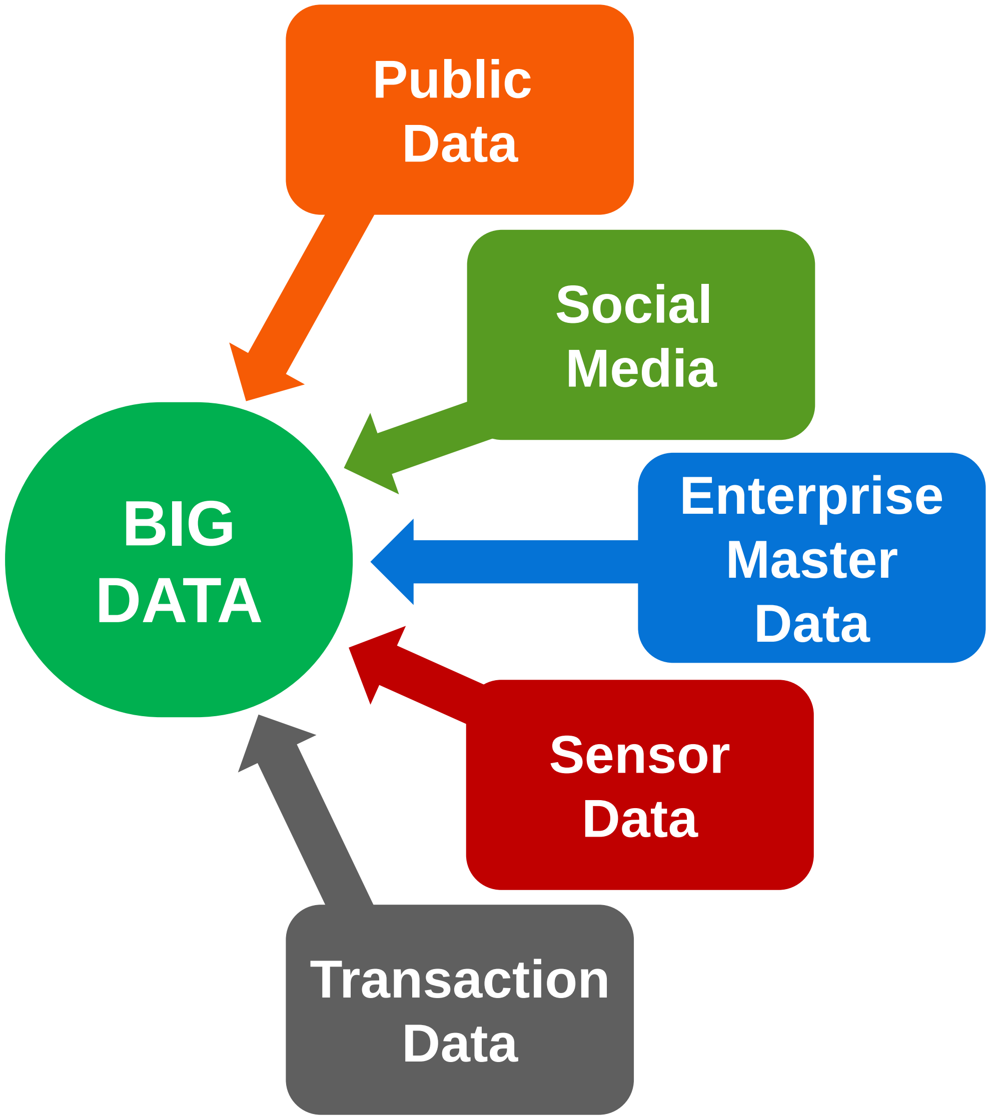
Factors Contributing Big Data Growth



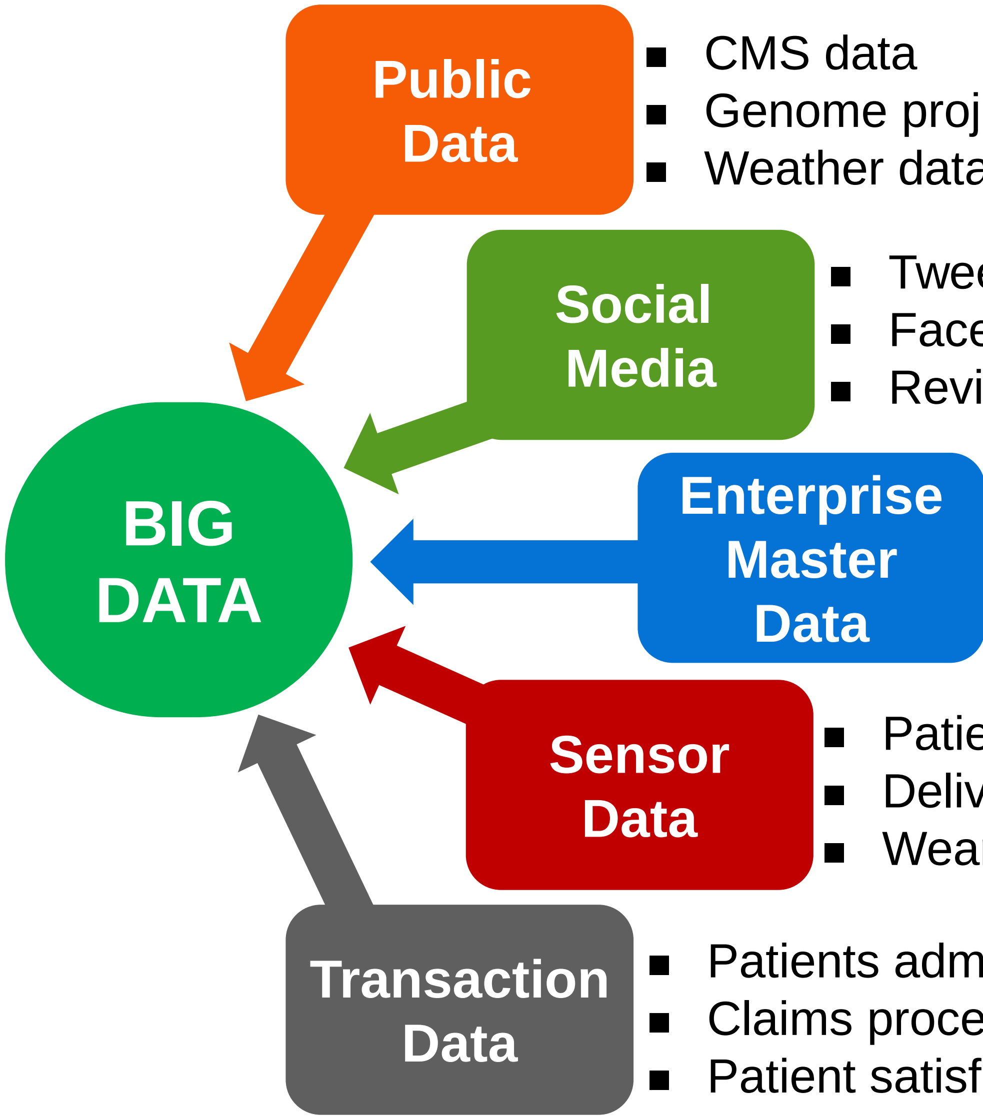
Emergence of Big Data



Big Data Sources



Big Data Sources: Health Care



Public Data
- CMS data
- Genome project
- Weather data

Social Media
- Tweets on epidemics
- Facebook comments on side effects
- Reviews on services

Enterprise Master Data
- Cost of services
- Provider details
- Employees
- Facilities

Sensor Data
- Patient vitals monitoring
- Delivery monitoring
- Wearable data

Transaction Data
- Patients admitted
- Claims processed
- Patient satisfaction

Big Data Sources: Retail

BIG DATA

Public Data
- Weather data
- Customer complaints
- US census data

Social Media
- Tweets on customer service
- Product recommendations on Facebook
- Product reviews

Enterprise Master Data
- Product inventory
- Cost or price of products
- Employees
- Customer information

Sensor Data
- Traffic measurement
- Weblog
- Wi-Fi traffic in stores

Transaction Data
- Customer purchases
- Financial accounting
- Inventory management
- Supply Chain

Examples of Companies Using Big Data

Yahoo

- >4500 nodes
- 60% of jobs run on Apache PIG
- Log processing, advertising analytics

Facebook

- >1000 nodes
- Hive-based data warehouse
- Reporting, analytics, and machine learning

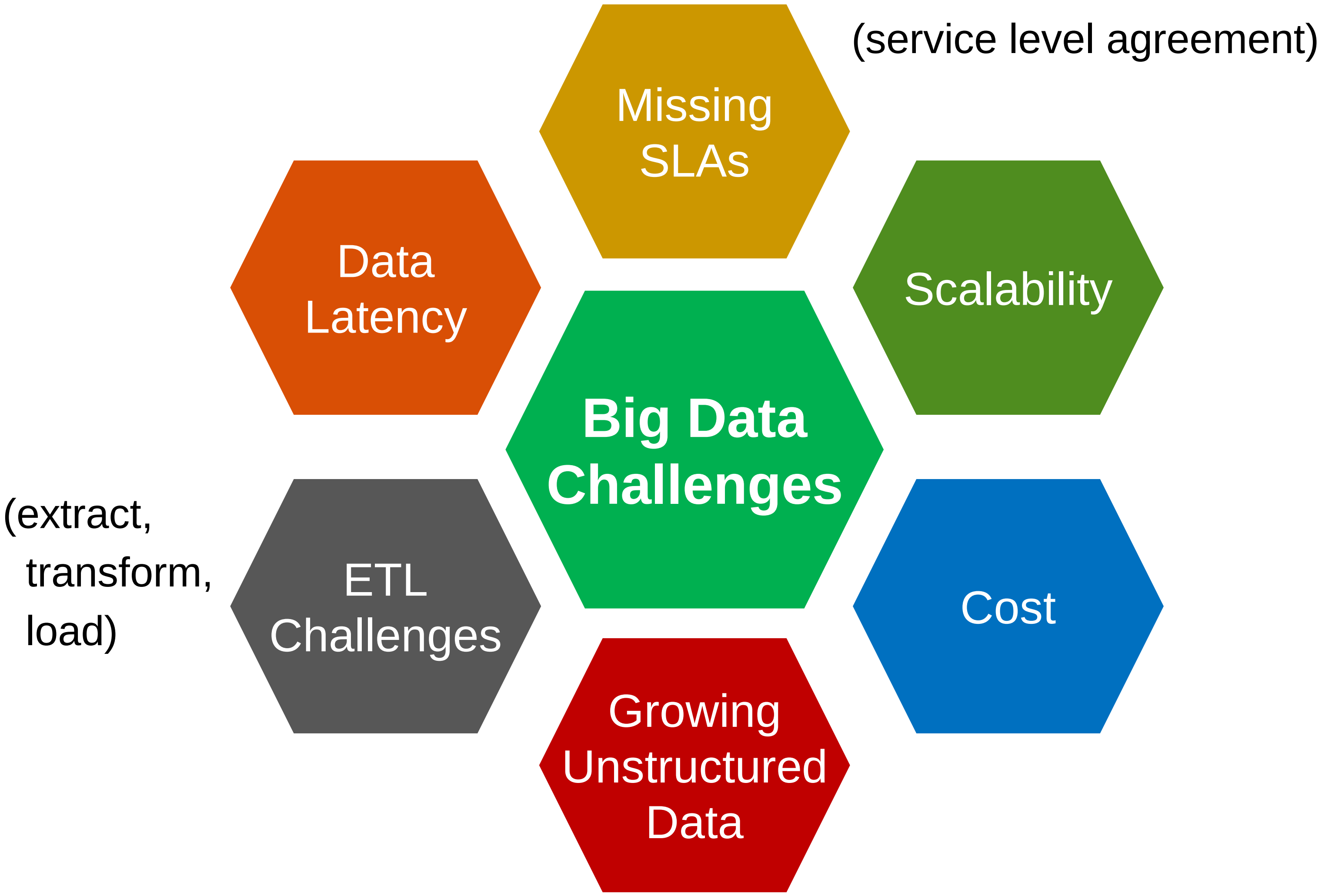
Enterprise Challenges Due to Big Data

Big data users encounter many challenges.

Big Data Challenges

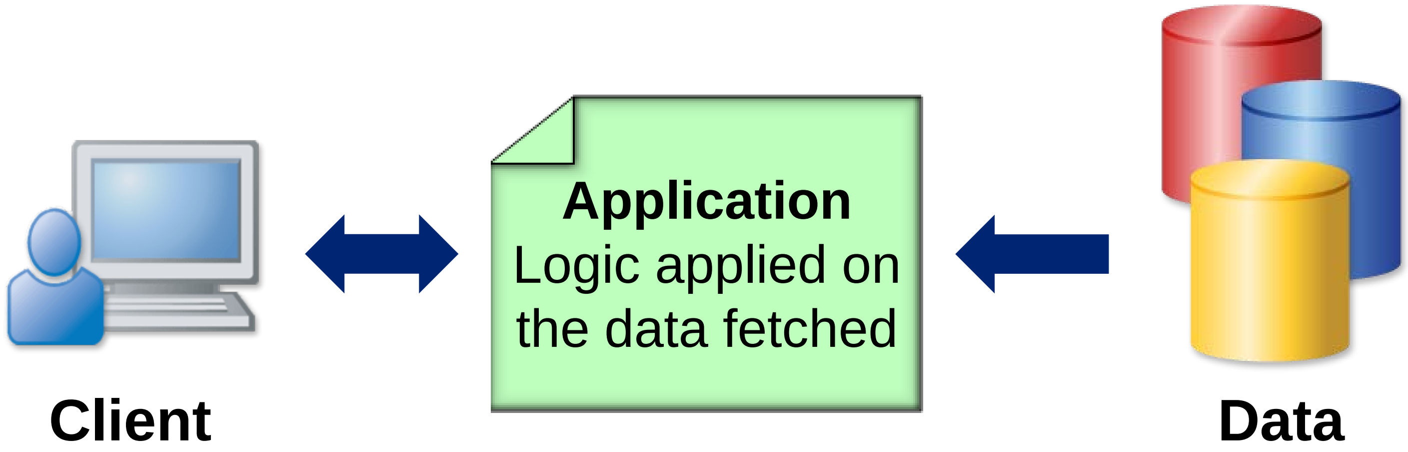
Enterprise Challenges Due to Big Data

Big data users encounter many challenges.



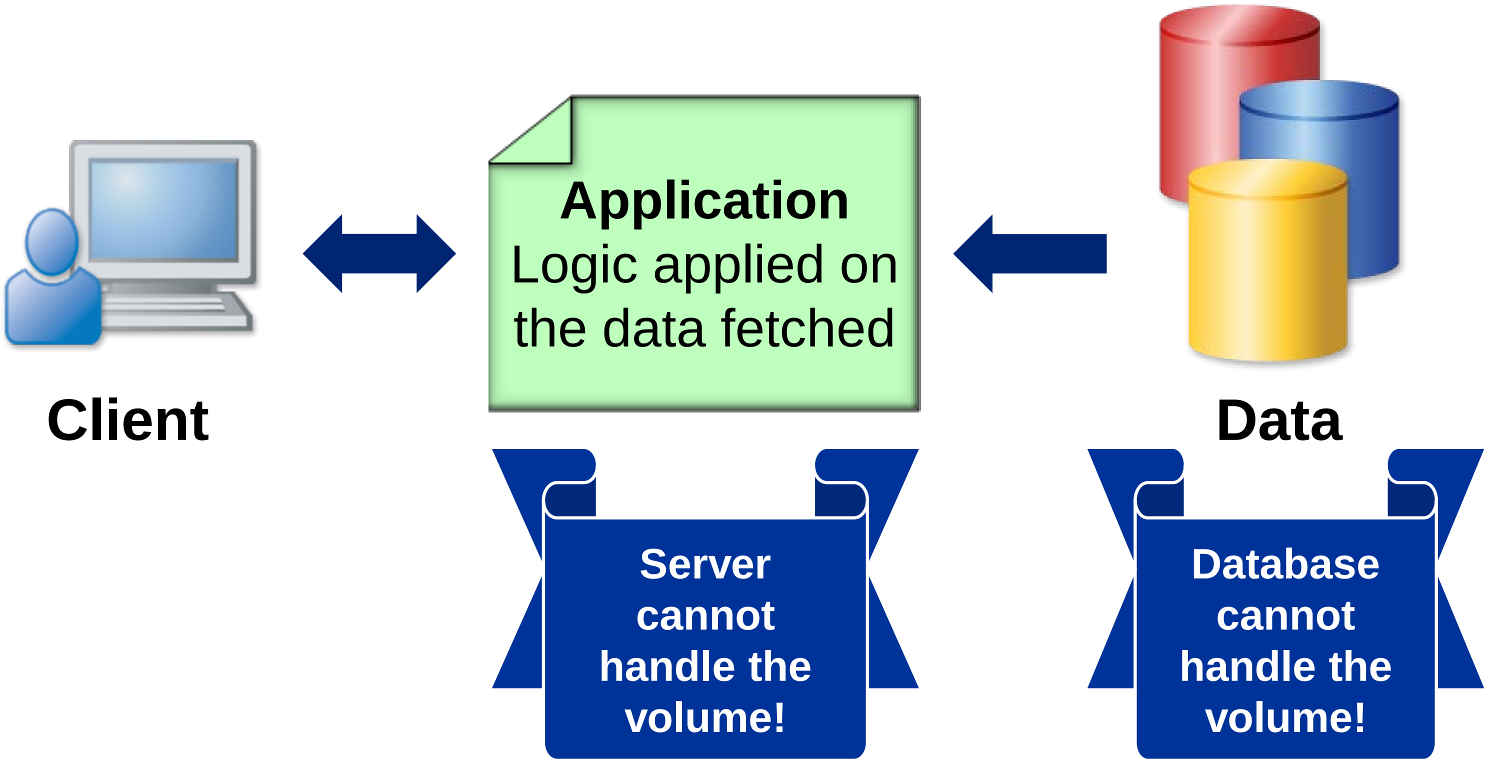
Traditional Data Processing: Architecture

In the traditional data processing model, data moves to the physical hardware that contains the application logic.



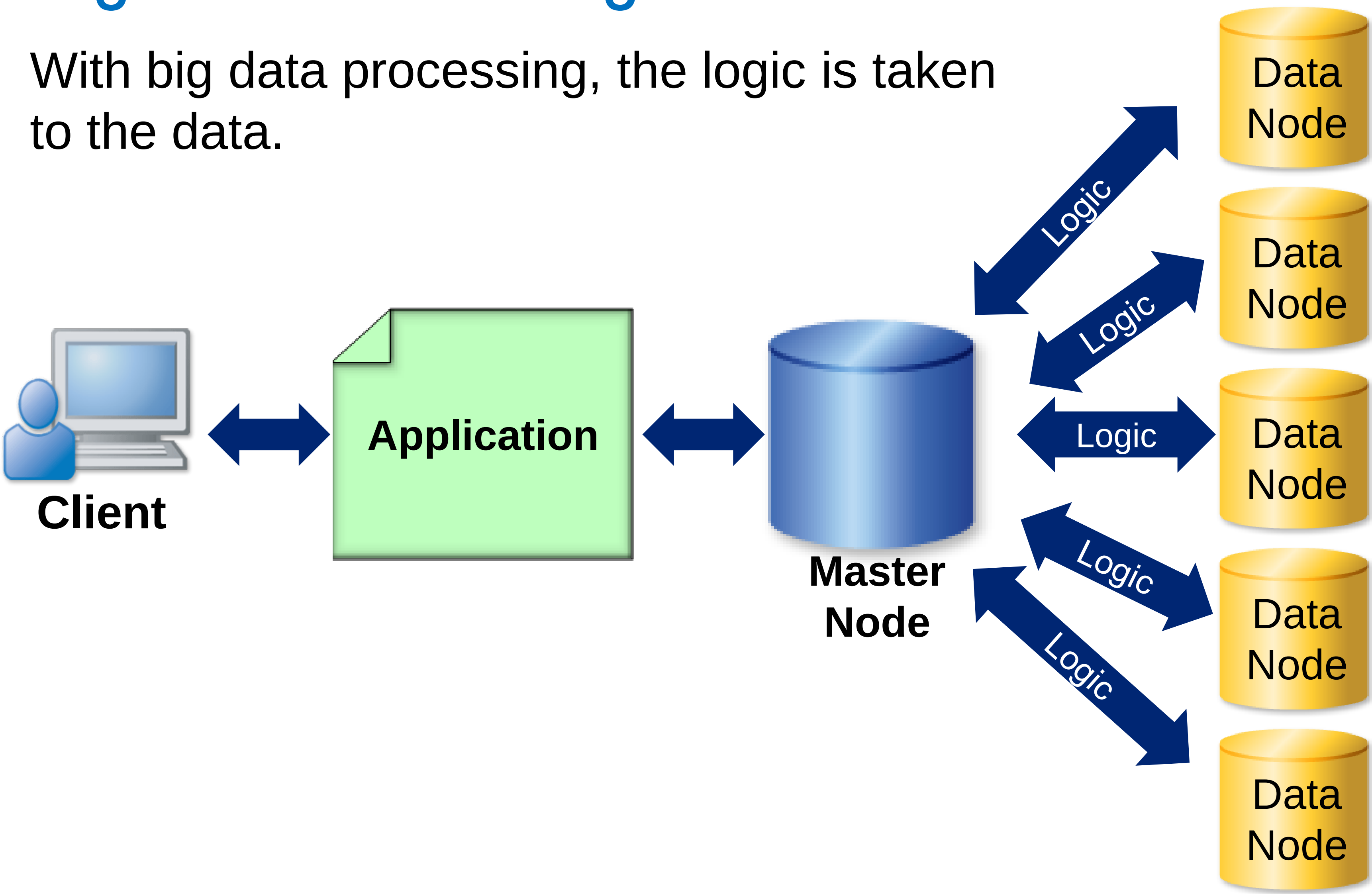
Traditional Data Processing: Big Data Impact

In traditional data processing models, with big data, it is possible that the application logic or hardware (or both) that it runs on cannot handle the volume.



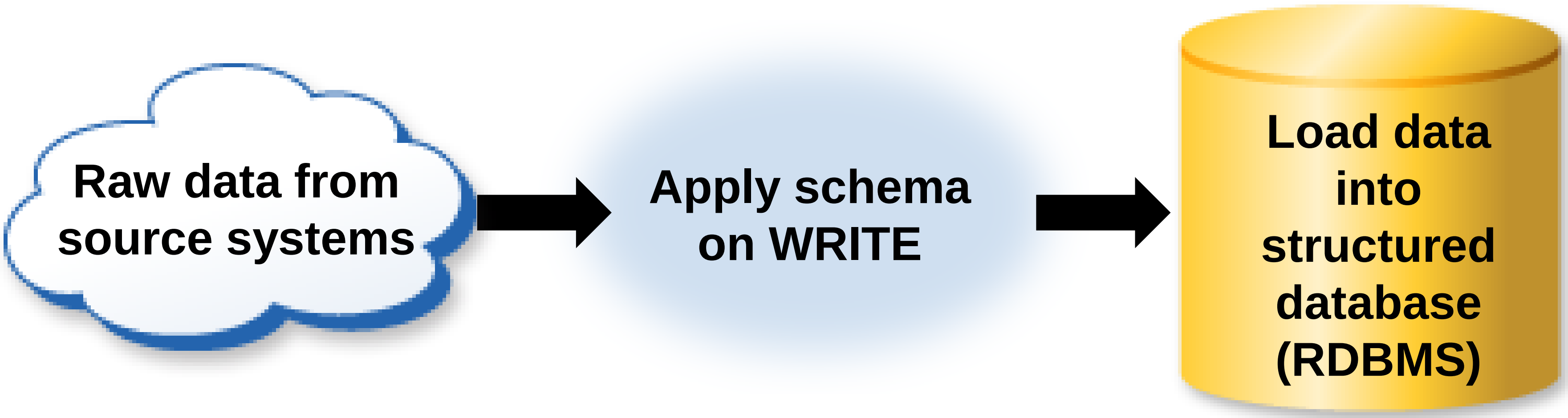
Big Data Processing: Architecture

With big data processing, the logic is taken to the data.



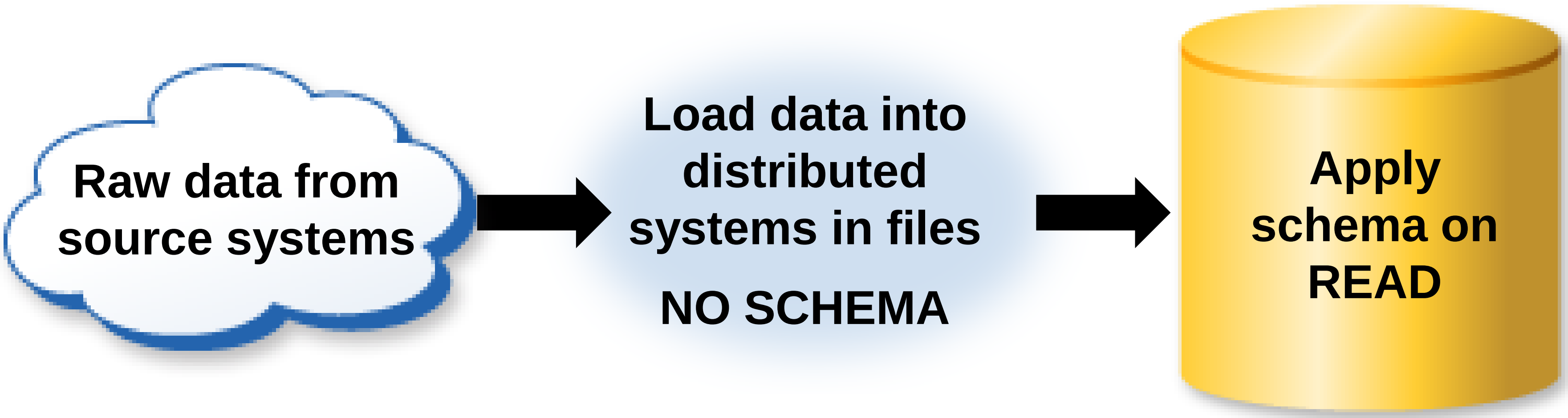
Traditional Data Management

In traditional data management, the data schema is applied on WRITE.

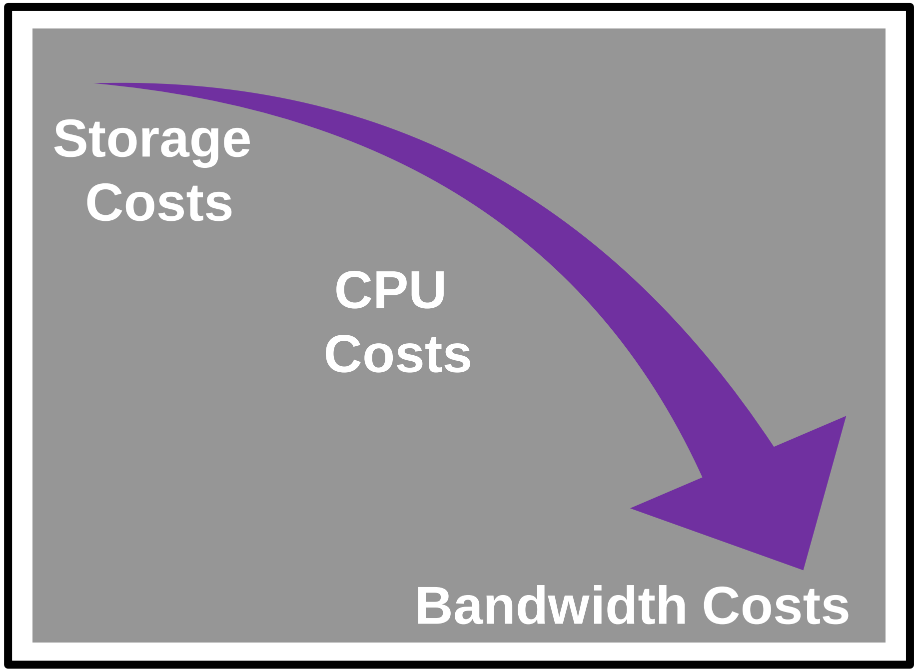


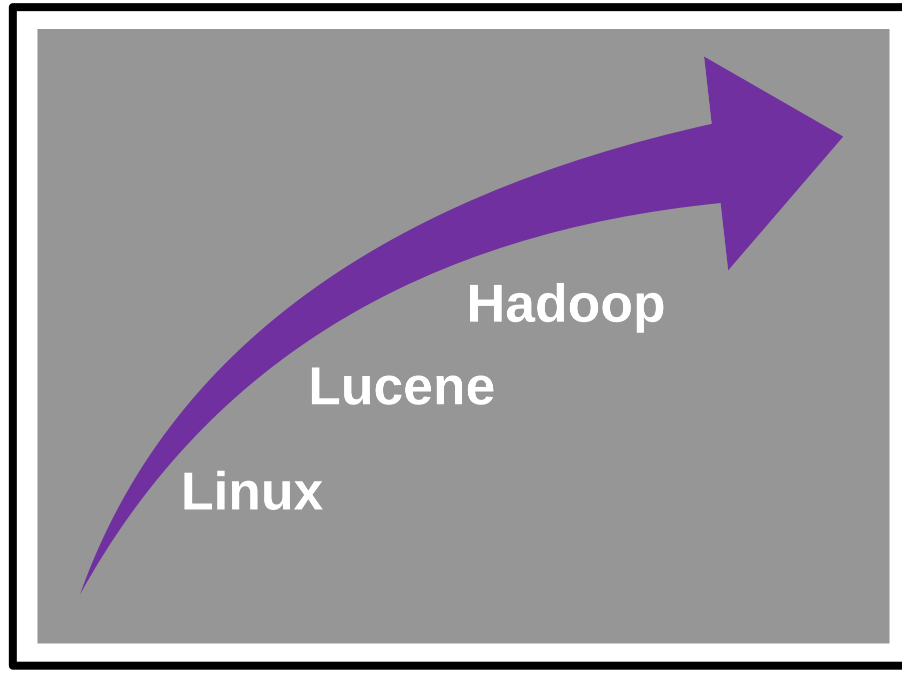
Big Data Management

In big data management, the schema is applied on READ.



Big Data Economics and Key Drivers





Hardware Cost Reduction

Open-Source Technologies

Data Processing Models

Two distinct data processing models can be followed:

- **Scale up** by adding bigger, more powerful processing machines.

- **Scale out** by taking advantage of existing hardware already in inventory, or by purchasing more of the smaller, less expensive machines.

Scale Up versus Scale Out

A decision to either scale up or scale out should be made. Compare costs versus hardware versus other considerations.

	Scale Up	Scale Out
Cost	Expensive	Cheap
Hardware	Specialized	Commodity
Fault Tolerance	Low	High
Licensing	Proprietary	Open source and proprietary options
Storage	Terabytes	Petabytes and more

Big Data versus Traditional Technologies

Traditional Data Systems	Big Data Technologies
Rigid data models	Schema free
Weak fault-tolerance architecture	Strong fault-tolerance architecture
Scalability constraints	Highly scalable
Expensive to scale	Economical (1TB ~ 5K)
Limitation for handling unstructured data	Can handle unstructured data
Proprietary hardware and software	Commodity hardware and open-source and proprietary software

Summary of Big Data Ecosystem

Big data ecosystems have the following characteristics:

- Data volumes are exploding.
- Volume, variety, and velocity vary greatly for big data.
- Hadoop is an example of a scale-out model.
- Decreasing commodity hardware costs make Hadoop a leading platform for big data.
- No schema, schema-less, and unstructured data can be handled.

The Apache Hadoop Project

Hadoop

Hadoop is a software framework with these capabilities:

- offers reliable, scalable, distributed computing
- enables the distributed processing of large data sets across clusters of computers using simple programming models
- designed to scale up from single servers to thousands of machines, each offering local computation and storage

See http://hadoop.apache.org.

History of Hadoop

Inventors: Doug Cutting and Mike Cafarella

Apache Software Foundation is non-profit

“Hadoop” is the name of Doug Cutting’s child’s yellow stuffed elephant

https://wiki.apache.org/hadoop/PoweredBy lists ***many*** companies that use Hadoop and what they use it for.

Major Releases – V1.x and V2.x

2005 – V1.x

- Hadoop Common – libraries and utilities
- Hadoop Distributed File System – (HDFS) distributed file system that stores data
- Hadoop Map Reduce – programming model

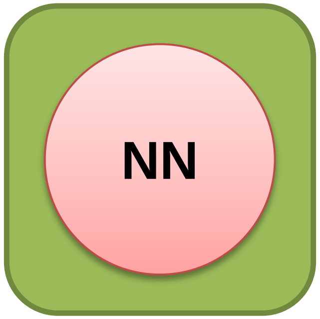
2013 – V2.x

- Hadoop Common
- Hadoop Distributed File System
- Hadoop YARN – resource management platform (**Y**et **A**nother **R**esource **N**egotiator)
- Hadoop Map Reduce

Hadoop – Name Node (NN)

In the Hadoop ecosystem, the Name node

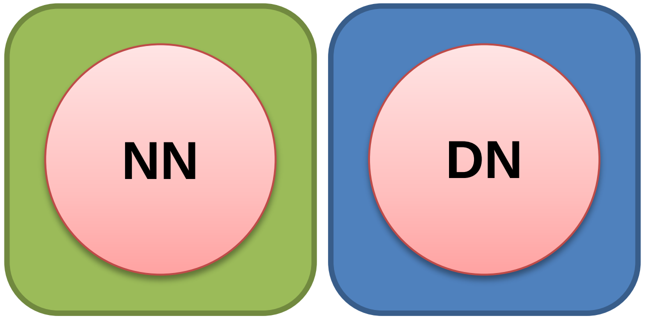
- is the centerpiece of an HDFS file system
- has RAM, CPU, and storage
- is usually more powerful (CPU, RAM) than a Data node
- stores HDFS file metadata to keep track of data in the HDFS directories across the Data nodes
- does not store the data
- can host a Job Tracker.



Hadoop – Data Node (DN)

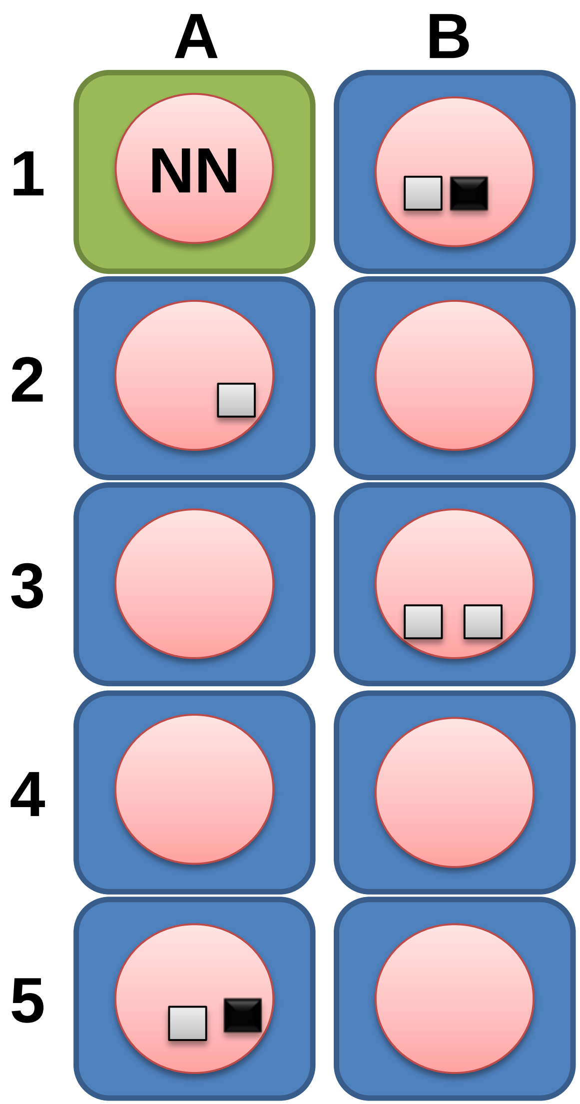
In the Hadoop ecosystem, the Data node

- has RAM, CPU, and storage
- stores and processes data locally
- has data that can be replicated to it (optional)
- hosts a Task Tracker.



The Name node and Data node should always be configured on separate machines.

Splits and Replication



- A file is split into small chunks (64MB, 128MB, and so on) and copied across the cluster.
- Data can be replicated. (optional)
- The recommended replication is 3 for production systems.

□ - 5 chunks A2, A5, B1, B3, B3

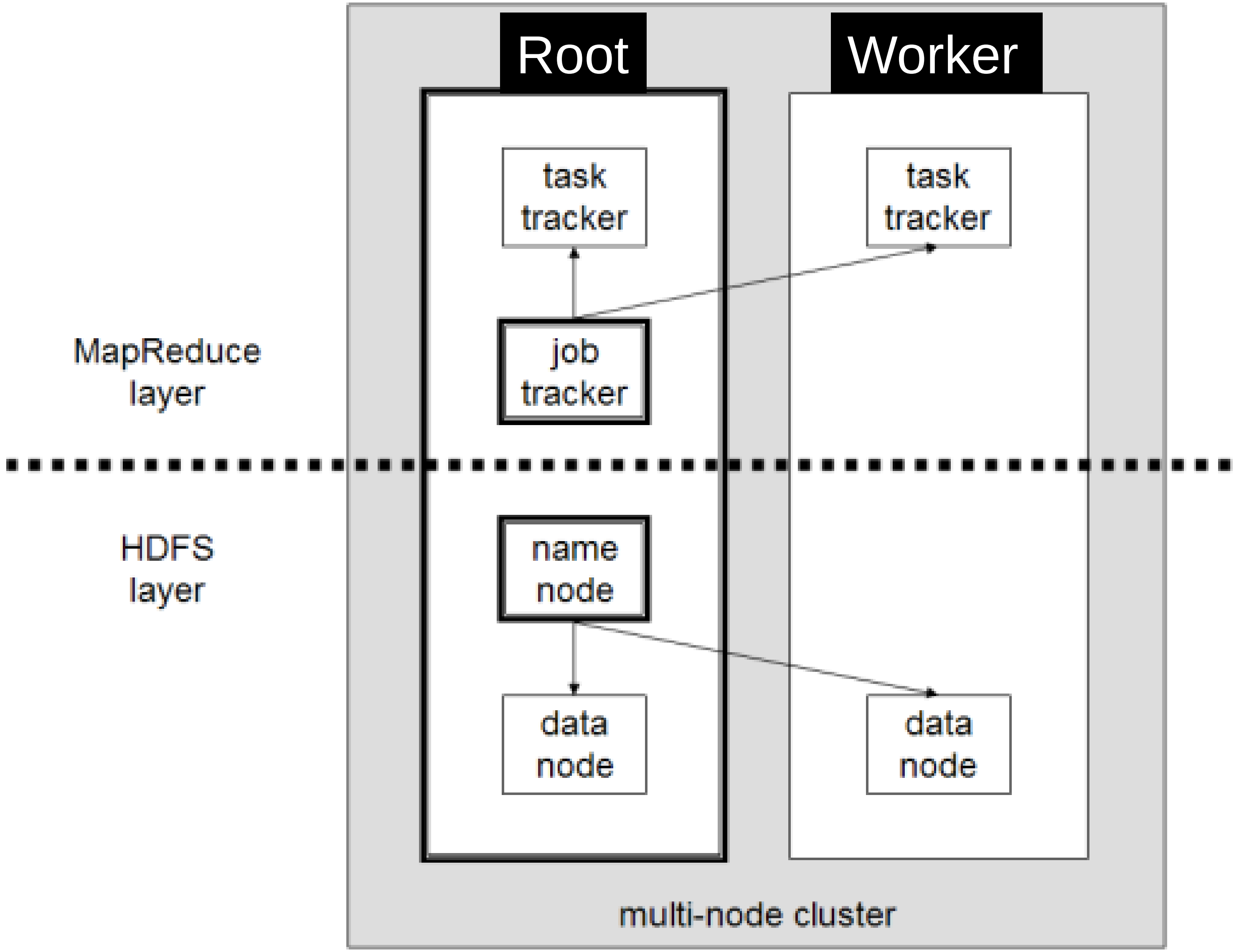
■ - 2 chunks A5, B1

Hadoop Data Processing Trackers

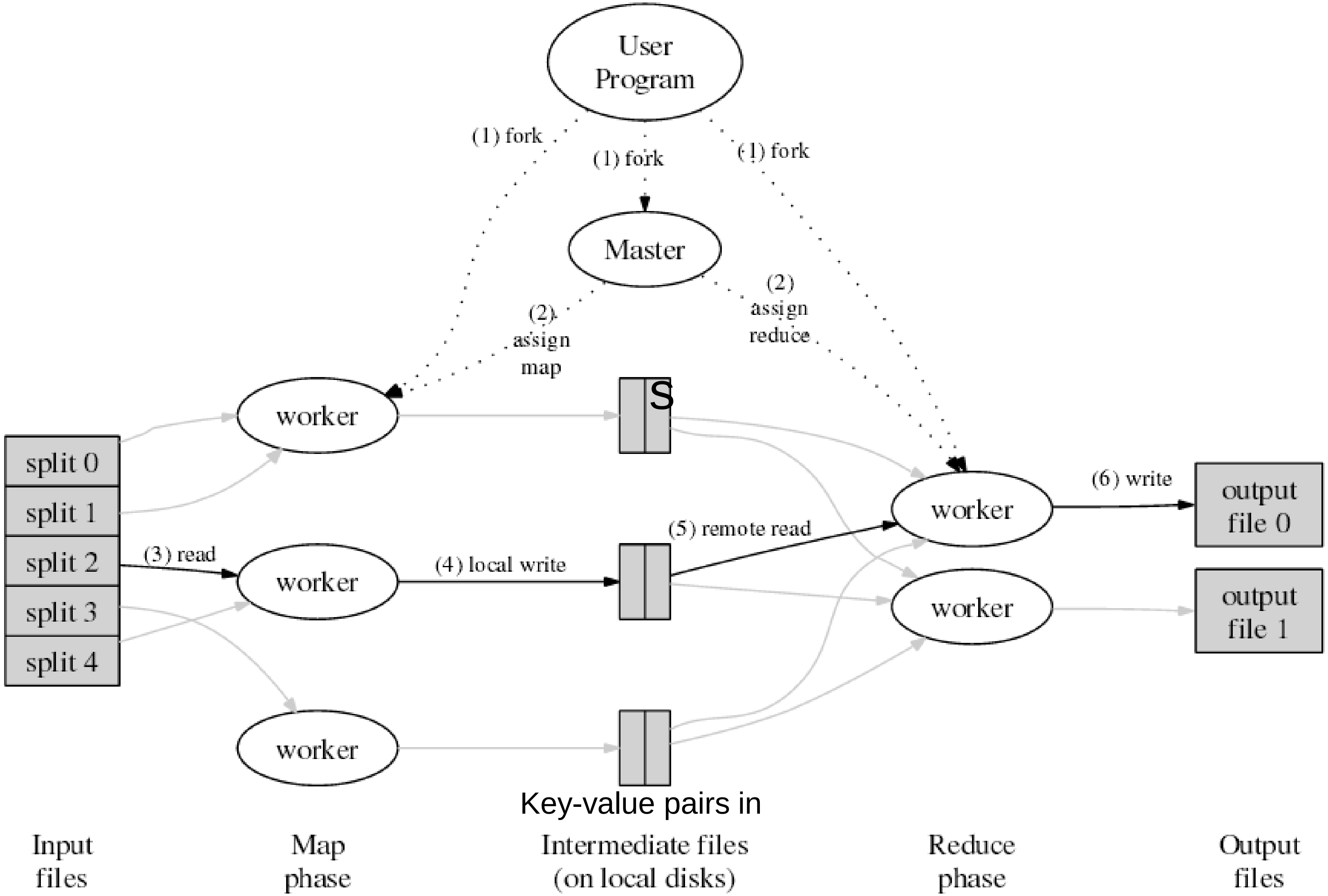
Hadoop Data Processing Trackers include the following:

- Job Tracker
  - manages resources and the Task Tracker
- Task Tracker
  - takes orders from Job Tracker
  - updates Job Tracker

Hadoop Data Processing – MapReduce



MapReduce Processing



Reference: http://code.google.com/edu/parallel/mapreduce-tutorial.html

Example from
https://courses.cs.washington.edu/courses/cse490h/08au/lectures/
MapReduceDesignPatterns-UW2.pdf

Pointer Following (or) Joining

Input

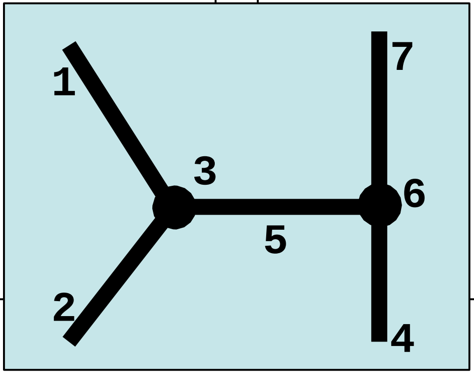
Feature List

```
1: <type=Road>, <intersections=(3)>, <geom>, …
2: <type=Road>, <intersections=(3)>, <geom>, …
3: <type=Intersection>, stop_type, POI? …
4: <type=Road>, <intersections=(6)>, <geom>,
5: <type=Road>, <intersections=(3,6)>, <geom>, …
6: <type=Intersection>, stop_type, POI?,  …
7: <type=Road>, <intersections=(6)>, <geom>, …
8: <type=Town>, <name>, <geom>, …
                    .
                    .
                    .
```

Output

Intersection List

```
3: <type=Intersection>, stop_type, <roads=(
     1: <type=Road>, <geom>, <name>, …
     2: <type=Road>, <geom>, <name>, …
     5: <type=Road>, <geom>, <name>, … )>, …
6: <type=Intersection>, stop_type, <roads=(
     4: <type=Road>, <geom>, <name>, … ,
     5: <type=Road>, <geom>, <name>, … ,
     7: <type=Road>, <geom>, <name>, … )>, …
                    .
                    .
                    .
```



Inner Join Pattern

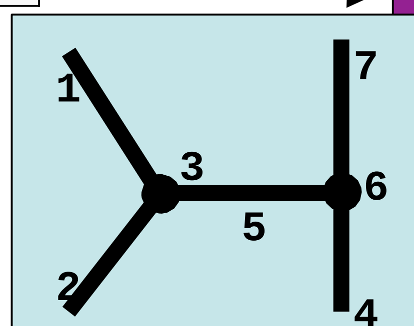
Input	Map	Shuffle	Reduce	Output
Feature list	Apply **map()** to each; Key = intersection id Value = feature	Sort by key	Apply **reduce()** to list of pairs with same key, gather into a feature	Feature list, aggregated

Input	Map
1: Road	(3, 1: Road)
2: Road	(3, 2: Road)
3: Intersection	(3, 3: Intxn)
4: Road	(6, 4: Road)
5: Road	(3, 5: Road)
6: Intersection	(6, 5: Road)
7: Road	(6, 6: Intxn)
	(6, 7: Road)

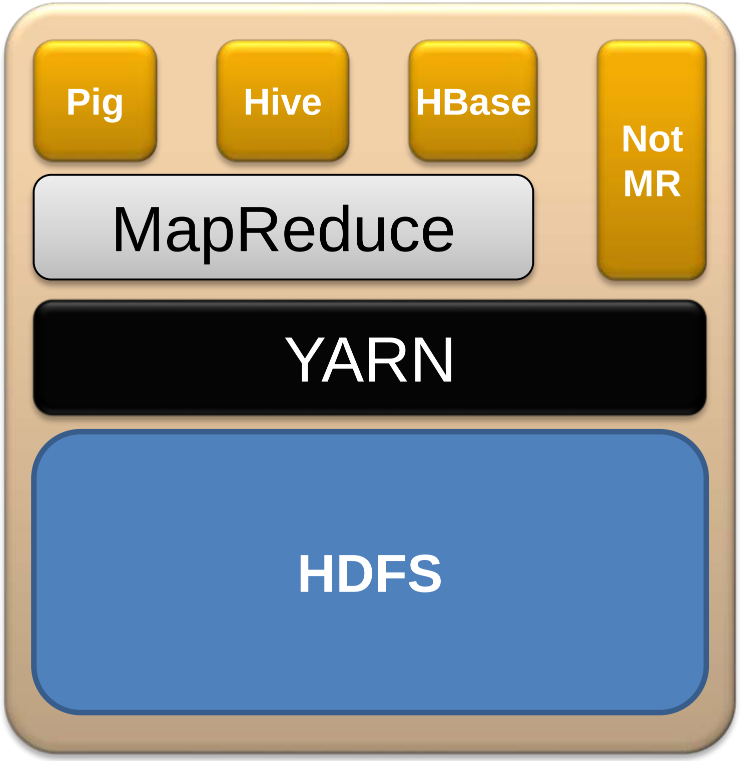
Key	Reduce input
3	(3, 1: Road) (3, 2: Road) (3, 3: Intxn.) (3, 5: Road)
6	(6, 4: Road) (6, 5: Road) (6, 6: Intxn.) (6, 7: Road)

3: Intersection
1: Road,
2: Road,
5: Road

6: Intersection
4: Road,
5: Road,
7: Road



YARN – Yet Another Resource Negotiator



Hadoop 2.x

YARN – Hadoop 2.x: Features

YARN .2x features include the following:

- multi-tenancy
  - not only batch-oriented
  - real-time applications
- cluster utilization
- scalability
- compatibility
  - backward compatible with Hadoop 1.x

Other members of the ecosystem

- **Apache Sqoop** is a framework designed for efficiently transferring bulk data between Hadoop and structured, relational databases (for example, MySQL and Oracle). It uses MapReduce to import and export the data.
- **Apache Pig** is a high-level platform for creating programs in Pig Latin that run on Hadoop. Pig can execute its jobs in MapReduce or Spark.
- **Apache Hive** is a data warehouse built on top of Hadoop for providing data summarization, query and analysis using an SQL-like interface.
- **Apache Spark** is an open-source cluster-computing framework. Typically it uses the Hadoop File System, but it is far more flexile than MapReduce. It is more RAM oriented than MapReduce (for better speed).